AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A computer-implemented method for buffering data in a multithreaded environment, comprising:

reading a buffer index value that identifies a data buffer that was last used for buffering data;

incrementing the buffer index value;

locating a buffer array entry that is associated with the buffer index value;

determining, at a particular time, whether the buffer array entry indicates a particular value;

if the buffer array entry does not indicate the particular value at the particular time, then, in response to a determination at the particular time that the buffer array entry does not indicate the particular value, attempting to obtain a lock on a particular data buffer that is associated with the buffer array entry; and

if the buffer array entry indicates the particular value at the particular time, then, in response to a determination at the particular time that the buffer array entry indicates the particular value, incrementing the buffer index value without attempting to obtain a lock on the particular data buffer.

2. (Previously Presented) The method of Claim 1, further comprising:

if the attempt to obtain the lock on the particular data buffer succeeds, then updating the buffer array entry to indicate the particular value.

3. (Previously Presented) The method of Claim 1, further comprising:

receiving a connection request from a client;

assigning a thread of execution to process said connection request; and

selecting a particular buffer management structure from a plurality of buffer management structures, wherein said plurality of buffer management structures are each associated with a set of data buffers that are used for buffering data to a physical memory unit;

wherein the buffer index value is associated with the particular buffer management structure.

4. (Previously Presented) The method of Claim 1, further comprising:

generating log data in response to a request for accessing a resource, wherein said resource represents one or more sets of content that are associated with a network server; and

selecting a buffer management structure based on one or more addresses in which said one or more sets of content are stored on said network server.

5-6. (Canceled)

7. (Previously Presented) The method of Claim 1, further comprising the step of writing log data into said particular data buffer.

8-9. (Canceled)

10. (Previously Presented) The method of Claim 1, further comprising:

maintaining a plurality of data buffers as an array of available buffers; and

in response to detecting that the particular data buffer contains a particular limited amount of free data space, removing said particular data buffer from said array of available buffers.

11. (Original) The method of Claim 10, wherein the step of removing said particular data buffer from said array of available buffers further comprises linking said particular data buffer into a list of ready-to-write data buffers.

12. (Original) The method of Claim 11, further comprising:
removing said particular data buffer from said array of available buffers; and
storing on a non-volatile storage unit information contained in said particular data buffer.

13. (Previously Presented) The method of Claim 1, further comprising:
maintaining a plurality of data buffers as an array of available buffers; and
in response to determining that no data buffer is available in said array of available buffers for storing said log data, requesting a free data buffer from a global list of free data buffers.

14-35. (Canceled)

36. (Currently Amended) A tangible computer-readable medium carrying one or more sequences of instructions for buffering data in a multithreaded environment, wherein execution of the one or more sequences of instructions by one or more processors causes the one or more processors to perform the steps of:
reading a buffer index value that identifies a data buffer that was last used for buffering data;
incrementing the buffer index value;
locating a buffer array entry that is associated with the buffer index value;
determining, at a particular time, whether the buffer array entry indicates a particular value;
if the buffer array entry does not indicate the particular value at the particular time, then, in response to a determination at the particular time that the buffer array entry does not indicate the particular value, attempting to

obtain a lock on a particular data buffer that is associated with the buffer array entry; and

if the buffer array entry indicates the particular value at the particular time, then, in response to a determination at the particular time that the buffer array entry indicates the particular value, incrementing the buffer index value without attempting to obtain a lock on the particular data buffer.

37. (Previously Presented) The computer-readable medium of Claim 36, further comprising instructions for performing the steps of:

if the attempt to obtain the lock on the particular data buffer succeeds, then updating the buffer array entry to indicate the particular value.

38. (Previously Presented) The computer-readable medium of Claim 36, further comprising instructions for performing the steps of:

receiving a connection request from a client;

assigning a thread of execution to process said connection request; and

selecting a particular buffer management structure from a plurality of buffer management structures, wherein said plurality of buffer management structures are each associated with a set of data buffers that are used for buffering data to a physical memory unit;

wherein the buffer index value is associated with the particular buffer management structure.

39. (Previously Presented) The computer-readable medium of Claim 36, further comprising instructions for performing the steps of:

generating log data in response to a request for accessing a resource, wherein said resource represents one or more sets of content that are associated with a network server; and

selecting a buffer management structure based on one or more addresses in which said one or more sets of content are stored on said network server.

40. (Previously Presented) The computer-readable medium of Claim 36, further comprising instructions for performing the step of writing log data into said particular data buffer.

41. (Previously Presented) The computer-readable medium of Claim 36, further comprising instructions for performing the steps of:
maintaining a plurality of data buffers as an array of available buffers; and
in response to detecting that the particular data buffer contains a particular limited amount of free data space, removing said particular data buffer from said array of available buffers.

42. (Previously Presented) The computer-readable medium of Claim 41, wherein the step of removing said particular data buffer from said array of available buffers further comprises linking said particular data buffer into a list of ready-to-write data buffers.

43. (Previously Presented) The computer-readable medium of Claim 42, further comprising instructions for performing the steps of:
removing said particular data buffer from said array of available buffers; and
storing on a non-volatile storage unit information contained in said particular data buffer.

44. (Previously Presented) The computer-readable medium of Claim 36, further comprising instructions for performing the steps of:
maintaining a plurality of data buffers as an array of available buffers; and

in response to determining that no data buffer is available in said array of available buffers for storing said log data, requesting a free data buffer from a global list of free data buffers.

45. (Currently Amended) A computer system, comprising:

means for reading a buffer index value that identifies a data buffer that was last used for buffering data;

means for incrementing the buffer index value;

means for locating a buffer array entry that is associated with the buffer index value;

means for determining, at a particular time, whether the buffer array entry indicates a particular value;

means for attempting to obtain a lock on a particular data buffer that is associated with the buffer array entry in response to a determination that the buffer array entry does not indicate the particular value at the particular time; and

means for incrementing the buffer index value without attempting to obtain a lock on the particular data buffer in response to a determination that the buffer array entry indicates the particular value at the particular time.